

COP 3502 - Computer Science I



- Announcement:
  - If you come to the exam late
  - And if at least one student has already left/finished
  - You will not be allowed to take the exam that day
  - And you will have to take a Makeup
  - So sounds like a good idea, right?
    - Just come late and you then get 2 more days to study!
    - For whatever reason, Makeup Exams are traditionally harder
    - Trust me, you do NOT want to do a makeup!



- Outline of Material Covered:
  - I. Arrays, Pointers, Strings, Files, Structs
    - Manipulation of array elements
    - Manipulation of struct components
      - Use of '.' vs '->'
    - Use of strings
      - strcmp, strlen, strcpy
    - Use of files (fopen, fscanf)
    - Dynamic memory allocation for arrays and for structs
      - malloc, calloc, and realloc



- Outline of Material Covered:
  - Linked Lists
    - Traversing a linked lists
      - Printing a list
      - Modifying list contents
    - How to allocate a node dynamically
    - Inserting elements anywhere in the list
    - Deleting elements anywhere in the list
    - You can be SURE to have at least one CODING question on linked lists
      - Everything is fair game including insert/delete.
      - Know the code!



#### Outline of Material Covered:

- III. Recursion
  - Fibonacci, Factorial, Binary Search
  - Writing recursive functions
  - Tracing through recursive functions
  - Towers of Hanoi
  - Permutation
  - Reversing a string
  - Also a good chance of having a recursion coding question



#### Outline of Material Covered:

- II. Algorithm Analysis
  - Big-O definition and finding the c value as shown in class
    - Understanding the various orders and what they mean
  - "Practical" Problems such as those on the slides and also during the lab
  - Analyzing code fragments and determining Big-O
  - Solving summations
  - Putting summations in their closed form (in terms of n)
  - Analyzing code fragments and using summations to determine the Big-O OR the specific number of a certain operation (multiplications, divisions, subtractions, etc.)



- How to study:
  - KNOW and UNDERSTAND the notes
  - Make sure you are 100% on the notes
  - Make sure you are 100% on all the lab questions and their respective solutions
  - Don't waste time memorizing algorithms
    - Understand how they work and WHY they work
    - And be prepared to come up with your own
  - Look at previous Foundation Exam tests
    - Practice some of the problems (ones that are applicable)
    - http://www.cs.ucf.edu/registration/exm/index.html



- Types of Questions:
  - Some short answer questions:
    - Tracing through code
    - Questions on an algorithm discussed in class
    - Small questions on code
    - Solve summations
    - Solve the "Practical" Problems
    - etc.
  - Writing Functions:
    - You will have to write functions
    - Almost guaranteed to be some recursive ones



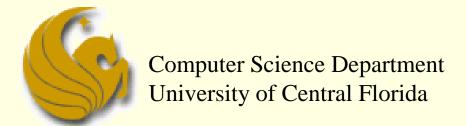
#### Exam Aids:

- You may use one 8-1/2"x11" sheet of paper
  - FRONT AND BACK
  - Typed or written doesn't matter
  - I don't care what you put on it
- What you CANNOT use:
  - Any electronic device:
    - Calculator, phone, ipad, you get the idea
  - If you are seen holding ANY electronic device, you will get 10 points off immediately! If you were cheating with that device, then the consequences are, of course, far worse.



- So what is covered?
  - EVERYTHING until now
  - Even if I didn't "cover" it during this review
  - Anything and everything that was taught or shown in class or in the labs is fair game.

# Questions:



COP 3502 - Computer Science I